Amendments to The Claims

This listing of claims will replace all prior versions and listings of claims in the pending application.

A. Listing of Claims

1. (Currently Amended) A computer-readable medium having computer-executable instructions for performing a method of using expressions to establish a relationship between properties in code executable by a browser running on a computer, each property defining a characteristic of an object, the method comprising:

    declaring the value of a first property as a scalar value; ~~and~~

    declaring the value of a second property as an expression, wherein the second property is

        a function of the first property<u>; and</u>

    <u>automatically changing the value of the second property in response to a change in the</u>

        <u>value of the first property.</u>

2. (Cancelled)

3. (Cancelled)

4. (Currently Amended) The computer-readable medium of claim ~~2~~ <u>1</u> wherein the computer-readable instructions are written in HTML code and define a document that forms a user interface, the computer-readable medium having further computer-executable instructions for: updating the document upon changing the value of the second property.

5. (Previously Amended) A computer-readable medium having stored thereon a data structure, the data structure including a plurality of nodes, comprising:

    at least one leaf node stored in memory, each leaf node containing a scalar property;

    at least one expression node stored in memory, the at least one expression node

        containing an expression written in a markup language and defining a dynamic

        property, the expression being a function of the scalar property, the expression for

2

programming formatting instructions;

at least one pointer stored in memory, the at least one pointer mapping a dependent/dependency relationship between the scalar properties and the expressions, wherein upon notification of a change in the value for one of the scalar properties, a browser executes the expressions dependent on the scalar property having a changed value.

6. (Original) The computer-readable medium of claim 5 wherein the data structure is formed by a browser.

7. (Original) The computer-readable medium of claim 6 wherein the markup language is

HTML.

8. (Original) The computer-readable medium of claim 6 wherein two or more pointers map two

or more scalar properties, respectively, to a single expression.

9. (Original) The computer-readable medium of claim 6 wherein two or more pointers map a

single scalar properties to two or more expressions, respectively.

10. (Original) The computer-readable medium of claim 6 wherein two or more pointers map a

single expression to two or more different expressions, respectively.

11. (Original) The computer-readable medium of claim 6 wherein at least one of the pointers

maps a dependent/dependency relationship between two expressions.

12. (Original) The computer-readable medium of claim 6 wherein:

one of the pointers maps a dependent/dependency relationship between a scalar property

and a first expression and another pointer maps a dependent/dependency

relationship between the first expression and a second expression, the first

expression forming a least-dependent expression and the second expression

forming a most-dependent expression; and

3

the browser executes the expressions in the order of the least-dependent expression to the most-dependent expression.

13. (Previously Presented) The computer-readable medium of claim 12 wherein:

two pointers map the second expression to the first expression and a third expression, respectively, the second expression being dependent on the first and third expressions thereby being more dependent than the first and third expressions; and

the browser does not execute the second expression until the first and third expressions are executed.

14. (Previously Amended) A method of using expressions to establish a relationship between properties of a document generated by HTML code that is executable by a browser, the properties including constant properties and dynamic properties, the method comprising:

declaring a value of one or more scalar properties, wherein the value of the scalar properties is not a function of another property;

creating one or more expressions declaring a value of a dynamic property, the dynamic property being a function of another property, the expression for programming formatting instructions; and

generating a dependency graph, the dependency graph having a plurality of nodes, including at least one expression node and one leaf node, the at least one expression node corresponding to an expression and the at least one leaf node corresponding to a value of a scalar property, the dependency graph mapping the relationship between the properties.

15. (Original) The method of claim 14 wherein the nodes have a dependent/dependency relationship to one another and each node has either a dirty state or a clean state, the method further comprising:

> when the value for one of the properties changes, changing the state of the node corresponding to the changed property to dirty;

> propagating the dirty state to each of the nodes in the dependency graph that are dependent on the node corresponding to the changed property; and

> recalculating the values of the expressions stored the nodes having a dirty state.

16. (Original) The method of claim 15 wherein recalculating the values of the expressions is performed automatically after completion of propagating the dirty state to each of the nodes in the dependency graph.

17. (Previously Presented) The method of claim 16 wherein:

> the dependency graph includes a plurality of expression nodes mapped in network wherein the most-dependent expression node in the network has no dependent expression nodes and the least-dependent expression node depends directly on a scalar property; and

> recalculating the values of expressions stored in nodes having a dirty state includes (a) executing the expression corresponding to each of the expression nodes in the network beginning with the least-dependent node and ending with the most-dependent node, and (b) assigning the value of each executed expression to the dynamic property corresponding to that executed expression.

5

18. (Original) The method of claim 17 wherein the HTML code generates a user interface upon execution and at the property declared by the expression corresponding to the child node defines as least a portion of the user interface, the method further comprising updating the user interface upon assigning the value to the property declared by the expression corresponding to the child node.

19. (Previously Amended) A computer for determining and outputting dynamic properties in HTML code received from a computer network, the computer comprising:

a processor:

a peripheral in data communication with the processor;

memory in data communication with the processor, the memory storing a browser for execution by the processor, the browser including code programmed to:

(a) analyze the HTML code to identify scalar properties and expressions, at least one of the expressions defining the value of a dynamic property and being a function of at least one scalar property;

(b) map a dependent/dependency relationship between the at least one of the expressions and the scalar properties, wherein each expression depends from at least one scalar property; and

(b) upon notification of a change in value of the scalar property, execute the at least one of the expressions that depends on the scalar property having a changed value.

20. (Original) The computer of claim 19 wherein:

the scalar properties and the expressions are mapped into a dependency graph and the browser is further programmed to assign a dirty state to each scalar property upon

6

notification of a change in the scalar property's value, and propagate the dirty

state from the scalar property to each expression that is dependent on the dirty

scalar property; and

the browser code programmed to execute each expression is further programmed to

execute only those expressions having a dirty state.

21. (Previously Amended) The computer of claim 20 wherein the browser code programmed to

execute each expression is further programmed to execute each expression in the dependency

graph in sequential order from a least-dependent expression to a most-dependent expression, the

least-dependent expression depending directly on a scalar property and the most-dependent

expression having no dependent expressions.

22. (Original) The computer of claim 20 wherein the computer further includes an input for

receiving a signal, and the browser further includes code programmed to notice when a signal

received by the input changes the value of a scalar property, and in response thereto

automatically execute each expression that depends on the scalar property having a value

changed by the input signal.

23. (Original) The computer of claim 19 wherein the peripheral is a computer monitor and the

browser is further includes code programmed to display an HTML document having dynamic

properties on the computer monitor.

24. (Previously Amended) A computer-readable medium having a browser for executing

HTML code received from a computer network having a computer, the browser comprising:

code programmed to analyze HTML code and identify scalar properties and expressions, at least one of the expressions defining the value of a dynamic property and being a function of the scalar property;

code programmed to map a dependent/dependency relationship between the at least one of the expressions and the scalar property; and

code programmed to notice a change in a value of the scalar property and to execute the at least one of the expressions in response to the change in value of the scalar property.

25. (Previously Amended) Apparatus for determining and processing dynamic properties of a document in HTML code received at a client computing system in a network of server and client computing systems, said apparatus operating in a browser at a client station, said apparatus comprising:

an expression engine identifying expressions and scalar properties received at the browser, at least one of the expressions for a document defining the value of a dynamic property and being a function of at least one scalar property;

a recalc engine mapping a dependent/dependency relationship between the at least one of the expressions and the scalar properties, wherein the at least one of the expressions depends from at least one scalar property;

the recalc engine upon notification of a change in value of the scalar property executing each expression that depends on the scalar property having a changed value; and

a script engine responsive to the recalc engine and generating a display of the document in accordance with the dynamic property.

26. (Original) The apparatus of claim 25 wherein the recalc engine comprises:

a map module mapping the scalar properties and the expressions into a dependency graph; and

8

a propagate module assigning a dirty state to a scalar property upon notification of a

change in the scalar property's value, and propagating the dirty state from the

scalar property to each expression that is dependent on the dirty scalar property;

and

an execute module executing each expression having a dirty state to provide the dynamic

property from the expression.

27. (Previously Amended) The apparatus of claim 26 wherein the execute module executes

each expression in the dependency graph in sequential order from a least-dependent expression

to a most-dependent expression, the least-dependent expression depending directly on a scalar

property and the most-dependent expression having no dependent expressions.

28. (Original) The apparatus of claim 26 further comprises:

an input for receiving a signal;

notice module detecting when a signal received by the input changes the value of a scalar

property, and generating a changed property notification; and

the execute module in response to the changed property notification executes each

expression that depends on the scalar property having a value changed by the

input signal.

29. (Currently Amended) A computer data signal embodied in a carrier wave readable by a

computing system and encoding a computer program of instructions for executing a computer

process using expressions to establish a relationship between properties of a document generated

by HTML code that is executable by a browser, the properties including constant properties and

dynamic properties, said computer process comprising:

9

providing a value of one or more scalar properties, wherein the value of the scalar

properties is not a function of another property;

creating one or more expressions declaring a value of a dynamic property, each dynamic

property being a function of another property, the expression for programming

formatting instructions; and

generating a dependency graph, the dependency graph having a plurality of nodes,

including at least one expression node and one leaf node, each expression node

corresponding to an expression and each leaf node corresponding to a value of a

scalar property, the dependency graph mapping the relationship between the

properties.

30. (Original) The propagated signal of claim 29 wherein the nodes have a

dependent/dependency relationship to one another and each node has either a dirty state or a

clean state, and the computer process further comprises:

when the value for one of the properties changes, changing the state of the node

corresponding to the changed property to dirty;

propagating the dirty state to each of the nodes in the dependency graph that are

dependent on the node corresponding to the changed property; and

recalculating the values of the expressions stored in nodes having a dirty state.

31. (Original) The propagated signal of claim 30 wherein the act of recalculating the values of

the expressions is performed automatically after completion of the act of propagating the dirty

state to each of the nodes in the dependency graph.

32. (Original) The propagated signal of claim 30 wherein the dependency graph includes a

plurality of expression nodes mapped in network wherein the most-dependent expression node in

the network has no dependent expression nodes and the least-dependent expression node depends directly on a scalar property; and the act of recalculating comprises:

executing the expression corresponding to each of the expression nodes in the network beginning with the least-dependent node and ending with the most-dependent node; and

assigning the value of each executed expression to the dynamic property corresponding to that executed expression.

33. (Original) The propagated signal of claim 32 wherein the HTML code generates a user interface upon execution and at the dynamic property declared by the expression defines at least a portion of the user interface, and the computer process further comprises: updating the user interface upon assigning the value to the dynamic property.